Online Learning and Reinforcement Learning
End-Semester Exam
Course Instructor : Arun Rajkumar.
Duration : Until 5 PM on 24th August, 2025

INSTRUCTIONS: Please submit a single PDF titled with your name and roll number clearly specified. All answers must either be legibly written and scanned or typed. For programming based questions, paste code and the necessary plots obtained along with clear explanations. Plagiarism/copying of any form will lead to disciplinary action.

(1) State the Doubling trick formally. Explain in your own words what problem it solves and show how. (5 points)

(2) Consider the usual online learning protocol with a finite set of d experts. Consider an adversary who is always consistent with the majority of k out of d experts in the class (assume $k < d$ and k is odd). What is the worst case number of mistakes for any algorithm for this problem? Can you think of an algorithm which achieves the same? (10 points)

(3) An *epsilon*-greedy strategy for the stochastic multi-armed bandits set up exploits the current best arm with probability $(1-\epsilon)$ and explores with a small probability ϵ. Consider a problem instance with 10 arms where the reward for the i-th $(i = 1, \ldots, 10)$ arm is Beta distributed with parameters $\alpha_i = 5, \beta_i = 5 * i$.

- Implement the *epsilon*-greedy algorithm and compare it with the performance of the UCB and the EXP-3 algorithm. (5 points)
- Comment on your observations about the regret plots obtained in the previous part. (2.5 points)
- If you vary ϵ, how does the regret change? (2.5 points)

(4) Consider the problem of online learning on the simplex Δ_d where d = 1000; At round t, you predict p_t and receive a vector z_t and suffer a loss of $p_t^T z_t$. Assume the adversary picks the vector z_t as the t-th row in the dataset *Dataset_Z*. Implement FTRL with quadratic and entropic regularization for this problem and plot the regret over time.
Now, consider the following algorithm which first picks and fixes a random 1000 dimensional vector R sampled uniformly from $[0, 1/\eta]^d$ and uses the following rule for prediction

$$p_{t+1} = \arg\min_{p \in \Delta_d} \sum_{i=1}^{t} \left(p^T (z_i + R)\right)$$

How would you choose η for this problem? For the value chosen, plot the regret bound for this algorithm as well. How does the regret bound compare with the previous two algorithms for this problem? Link to Dataset
(10 points)